



# Guillaume FENOLLAR

## Ingénieur Linux / DevOps

Nouméa

Né le 02/01/1989

Célibataire

Permis B

### Contact

Tél : (+687) 539665

Email : [guillaume@fenollar.fr](mailto:guillaume@fenollar.fr)

Skype : [guillaume.fenollar](https://www.skype.com/people/guillaume.fenollar)

Blog : [guillaume.fenollar.fr](http://guillaume.fenollar.fr)

Gitlab : [guillaume.fenollar](https://gitlab.com/guillaume.fenollar)



### Diplômes

- 2011 : Master 1 en ingénierie des systèmes et réseaux (Orsay)
- 2009 : License de sécurité des services et systèmes informatiques (Orsay – Paris XI)
- 2008 : BTS Informatique de Gestion, option administration (Lycée Vilgénis – Massy)
- 2006 : Baccalauréat STI option électrotechnique (Lycée Kastler – Dourdan)



### Expériences professionnelles

Depuis mai 2018 : Freelance SysAdmin/DevOps/WebDev (Indépendant)

2017 – 2018 : DevOps détaché à la SLN (BBS)

- Gestion de l'infrastructure DevOps de la SLN (75 machines, ~700 conteneurs Docker)
- Mise en place des bonnes pratiques d'administration, de sécurité et de supervision

2014- 2017: Administrateur Systèmes Linux (Ubisoft)

- Gestion des systèmes dédiés au développement des jeux
- Tech lead de diverses technologies (Hansoft, ElasticSearch, TestRail...)
- Responsable de tous les environnements Web pour les productions EMEA
- Programmeur python d'outils et d'APIs pour les équipes IT

2009 – 2014 : Administrateur Systèmes Linux (XWiki)

- Gestion globale de l'infrastructure cliente (200 VMs) et du réseau local
- Installation et optimisation du stack Apache/Tomcat/ XWiki et des outils associés
- Contact client et support sur place (infrastructures locales)



### Projets Notables

**Evolution et stabilisation d'une infrastructure DevOps complète (BBS / SLN)**

*technologies utilisées : Docker, Gitlab, Artifactory, Kafka, ElasticSearch, Salt, Fabric...*

- Situation antérieure : Une infrastructure de développement sans réel outil ni méthode automatisée de gestion des environnements ni intégration continue.
- Multiplication des environnements (recettes, développement, intégration...) et gestion grâce à la mise en place d'applicatifs appropriés, notamment Configuration Management et Supervision.
- Réalisation de pipelines de déploiements (CI/CD) et de bonnes pratiques de développement (Gitlab).

**Mise en place d'une infrastructure CI (Ubisoft)**

*technologies utilisées : GitLab, Docker, Chef, CentOS 7*

- Situation antérieure : Les productions dans le besoin ne pouvaient disposer d'une solution clé en main pour gérer le CI (continuous integration) de leur projets.
- La mise à niveau complète de l'infrastructure vieillissante GitLab a débouché sur un projet plus complet de mise à disposition d'un pipeline complet de tests et de builds via GitLab CI
- Docker a été choisi pour la création à la volée de machines de build (runners). Une production n'a alors plus à s'occuper de créer sa propre infrastructure de builds et ses images.
- Le tout est automatisé avec Chef, y compris l'installation automatisée des mises à jour selon conditions.
- Le projet a été effectué avec l'aide d'un gestionnaire d'application expert en *source-control* (pour les choix non techniques).

## Programmation DevOps pour l'infrastructure Perforce (Ubisoft)

*technologies utilisées : Python 3.4, Perforce API, Flask, MySQL, CentOS 6*

- Beaucoup de tâches étaient faites à la main, ou pas optimisées, dans l'administration de l'infrastructure Perforce (une des plus importantes au monde, avec des machines de 1TB de RAM). J'ai pu donc écrire des programmes sur-mesures pour répondre à ces besoins :
- 1. Script de test synthétiques pour le monitoring (Zabbix). L'idée était de reproduire une action utilisateur complète pour détecter les erreurs le plus fidèlement et rapidement possible. Le script renvoie 12 codes d'erreurs différents pour pallier à tous les états de l'instance.
- 2. Programme de vérification continu de l'état des fichiers de tous les dépôts Perforce. Ceci permet de détecter la corruption des fichiers déposés sur le SAN et également d'être proactif sur les problèmes d'instabilité de bases de données. Le programme est installé en tant que démon, et fait la part belle à l'utilisation du multiprocessing ainsi qu'à une consommation mémoire optimisée.
- 3. Script de suppression des données intelligent, pour éviter l'utilisation en direct de la commande par défaut de suppression, trop gourmande en ressources (p4 obliterate). Il s'agit ici de s'adapter à la taille de l'instance Perforce, et de procéder par chunks pour limiter l'overhead sur l'application.
- 4. Réécriture du système d'inventaire des instances Perforce grâce à la mise au point de scripts de reporting et de leurs distributions, ainsi qu'au développement d'une API basée sur Flask, avec MySQL comme backend. Outil désormais utilisé pour toute la refacturation des instances aux productions de jeux en interne. Seul programme sur lequel je n'ai pas travaillé seul, mais en équipe de deux.

## Système d'inventaire et de supervision (XWiki)

*technologies utilisées : XWiki, Glimpse, Python 2.6, Debian 7*

- Glimpse est un système d'inventaire de machines virtuelles et physiques codé en python et Java, et développé avec l'aide d'un chercheur R&D pour la partie Java.
- L'affichage des ressources du réseau est directement remonté dans le wiki d'entreprise (XWiki).
- Tout type d'élément peut alors être remonté via l'écriture de scripts python à la volée.
- Le système de supervision Nagios génère alors automatiquement et continuellement sa configuration à partir de ces données.



## Compétences

- **Systemes d'exploitation :**
  - CentOS, RedHat, Debian, ArchLinux
  - Windows 2003/2008
- **Applicatifs :**
  - Virtualisation : Docker, Kubernetes, VMWare (VSphere)
  - Web : Nginx, Unicorn, Tomcat, HAProxy, GitLab, TestRail, Jira, XWiki
  - Données : MySQL, PostgreSQL, Kafka, Elasticsearch, Redis, Zookeeper
  - Automatisation : Salt, Chef, Puppet, CfEngine
  - Monitoring : Zabbix, Centreon, Nagios
- **Coding :**
  - Gestions de versions : Git, Perforce, SVN, CI (Jenkins & Gitlab CI)
  - Scripting : Python, Bash
  - Frameworks Python : Django, Flask, Fabric
  - Web : HTML, CSS3, Javascript, React
- **Linguistique :**
  - Français : langue maternelle
  - Anglais : bon niveau



## Centres d'intérêt

- Musique (musicien et producteur)
- Développement de sites Internet
- Jeux vidéo et pop-culture
- Vulgarisation scientifique et historique